

WHAT IS CLAIMED IS:

1. A method of providing a virtual file system including application data files to selected set-top terminals within a cable network, each of the selected set-top terminals having one or more client applications residing thereon, the method comprising the steps of:

creating a plurality of virtual streams in a single downstream service within a stream set where the virtual streams carry multicast addresses associated with the selected set-top terminals in the cable network;

streaming application data files from a data carousel onto one or more of the virtual streams; and

delivering the stream set to the client application in accordance with delivery criteria set by an API residing on the set-top terminal.

2. The method of claim 1 wherein the delivery criteria includes configuration of the virtual file system across the stream set.

3. The method of claim 2 wherein the configuration comprises partitioning.

4. The method of claim 1 wherein the delivery criteria include in-band or out-of-band transport criteria.

5. The method of claim 1 wherein the delivery criteria include bandwidth availability on the cable network.

- 
6. The method of claim 1 wherein the delivery criteria include bit-rate.
  7. The method of claim 1 wherein the stream set carries operations information.
  8. The method of claim 7 wherein the operations information includes structured information describing the virtual file system.
  9. The method of claim 8 wherein the structured information includes an MPEG-2 PAT.
  10. The method of claim 8 wherein the structured information includes an MPEG-2 PMT.
  11. The method of claim 1 wherein the stream set is MPEG-2 compliant.
  12. The method of claim 1 wherein the API includes wrapper functions that are provided to the client application.
  13. The method of claim 1 wherein the wrapper functions include a synchronous function-call and response.

14. The method of claim 1 wherein the application data files are streamed according to a file selection algorithm.
15. The method of claim 14 wherein the file selection algorithm selects files in a virtual directory.
16. The method of claim 14 wherein the file selection algorithm selects files that are applicable to time window.
17. The method of claim 14 wherein the file selection algorithm limits selected files to those which are applicable to a sliding time window.
18. An carousel file server for use in cable network applications, comprising:  
a user interface;  
a carousel manager operably coupled to exchange status and control information with the user interface;  
a staging area operably coupled to the carousel manager for temporarily storing uploaded application data files to the carousel manager; and  
a persistent data store operably coupled to the carousel manager for storing application data files and configuration data from the staging area under control of the carousel manager,  
wherein the carousel manager is arranged to transmit the application storage files repeatedly in a plurality of virtual streams within a single PID stream.

Docket: D2505

19. The carousel file server of claim 18 wherein the user interface comprises a web-browser operably coupled to a web server.

20. The carousel file server of claim 18 wherein the persistent data store further stores metadata.

21. The carousel file server of claim 18 wherein the persistent data store further stores server logs.

22. The carousel file server of claim 18 wherein the persistent data store comprises an SQL database.

23. The carousel file server of claim 18 wherein the carousel manager further includes a data streamer for obtaining and transmitting application data files.

24. The carousel file server of claim 23 wherein the data streamer comprises a data source and an interface handler.

25. The carousel file server of claim 18 wherein the carousel file manager is adapted to monitor the staging area for new or modified application data files.

26. The carousel file server of claim 25 wherein the carousel manager uploads the new or modified application data files to the persistent data store subsequent in response to the monitoring.

27. The carousel file server of claim 18 wherein the user interface, carousel manager, staging area and persistent data store are implemented using computer code embodied in computer-readable medium and resident on a programmable application server.

28. A method for providing an application programming interface (API) that is resident on a set-top terminal coupled to a cable network with virtual file mounting and directory information gathering functionality within a carousel file environment, the method comprising the steps of:

providing a plurality of wrapped code to assemble a set of related application data files from a plurality of virtual data streams or a stream set;

providing one or more wrapper functions to a client application running on the set-top terminal to access the wrapped code; and

executing a wrapped code in response to a wrapper function call placed by the client application.

29. The method of claim 28 wherein the wrapper functions are implemented using a synchronous function-call and return methodology.

30. The method of claim 28 wherein an asynchronous response to a function-call is implemented for a selected function.

Docket: D2505

31. The method of claim 30 wherein the selected function includes client application data file retrieval.
32. The method of claim 30 wherein the API is accessed through an API server resident on the set-top terminal.
33. The method of claim 30 wherein the API is resident in firmware in the set-top terminal.
34. The method of claim 30 wherein the API is downloaded to the set-top terminal as executable code.